

Clean copy of allowed claims

30. A computer implemented method that comprises a database system, to simulate effect of a recovery time required for the database system to recover from a database failure on runtime performance of the database system, the method comprising:

providing in the computer, in addition to a normal checkpoint queue used in the database system for normal operation, at least one simulated checkpoint queue;

wherein the normal checkpoint queue comprises a plurality of buffers;

wherein the simulated checkpoint queue is an ordered list of one or more elements, each element in the simulated checkpoint queue representing a respective buffer that is or was in the normal checkpoint queue, the ordered list having a head and a tail;

wherein the simulated checkpoint queue is associated with a setting for recovery time whose effect on runtime performance of the database system is being simulated in the computer;

in response to detecting a change to a buffer in the normal checkpoint queue due to actual database transactions occurring within the database system under normal operating conditions, checking if the buffer is represented in the simulated checkpoint queue, and if the buffer is not represented in the simulated checkpoint queue, linking an element that represents the buffer to the tail of the simulated checkpoint queue;

providing a simulated write counter, the simulated write counter being associated with the setting for recovery time;

wherein the simulated write counter provides a count of a number of times any element is removed from the simulated checkpoint queue in response to a simulated write out of the respective buffer from volatile memory to nonvolatile memory;

determining if linking the element to the simulated checkpoint queue causes the simulated checkpoint queue to exceed a predetermined length; and

in response to determining that the simulated checkpoint queue exceeds the predetermined length, removing an element from the head of the simulated checkpoint queue and incrementing the simulated write counter.

31. (Deleted).

32. (Deleted).

33. (Deleted).

34. The method of Claim 32, wherein the predetermined length is a dirty buffer limit.

35. The method of Claim 30 further comprising:
writing out of the any buffer from volatile memory and storing in nonvolatile memory using an incremental checkpoint operation in the normal operation of the database;
in response to detecting a write out of any buffer by the incremental checkpoint operation, checking if that buffer is represented in the simulated checkpoint queue, and if that

buffer is represented in the simulated checkpoint queue, removing the element representing that buffer from the simulated checkpoint queue and incrementing the simulated write counter.

36. (Deleted).

37. The method of Claim 30, wherein each element in the simulated checkpoint queue comprises:

a first identifier that identifies the respective buffer in the normal checkpoint queue;
and

a second identifier that identifies a journal entry in a redo log, the journal entry corresponding to the respective buffer.

38. The method of Claim 30, wherein the elements in the simulated checkpoint queue are ordered according to each element's journal entry position in a redo log in the database system.

39. The method of Claim 30 further comprising determining a dirty buffer limit for the simulated checkpoint queue, the dirty buffer limit specifying the length of the simulated checkpoint queue, the dirty buffer limit being determined from the setting of the recovery time, and historical operating data.

11. A computer-readable storage medium having stored thereon computer instructions that, when executed by a computer, cause the computer to simulate effect of a recovery time required for the database system to recover from a database failure on runtime performance of the database system, the instructions comprising instructions to:

provide in the computer, in addition to a normal checkpoint queue used in the database system for normal operation, at least one simulated checkpoint queue;

wherein the normal checkpoint queue comprises a plurality of buffers;

wherein the simulated checkpoint queue is an ordered list of one or more elements, each element in the simulated checkpoint queue representing a respective buffer that is or was in the normal checkpoint queue, the ordered list having a head and a tail;

wherein the simulated checkpoint queue is associated with a setting for simulated mean time to recover (MTTR) whose effect on runtime performance of the database system is being simulated in the computer;

in response to detecting a change to a buffer in the normal checkpoint queue due to actual database transactions occurring within the database system under normal operating conditions, check if the buffer is represented in the simulated checkpoint queue, and if the buffer is not represented in the simulated checkpoint queue, link an element that represents the buffer to the tail of the simulated checkpoint queue;

provide a simulated write counter, the simulated write counter being associated with the simulated MTTR setting;

wherein the simulated write counter provides a count of a number of times any element is removed from the simulated checkpoint queue in response to a simulated write out of the respective buffer from volatile memory to nonvolatile memory;

determine if linking the element to the tail of the simulated checkpoint queue causes the simulated checkpoint queue to exceed a predetermined length; and

in response to determining that the simulated checkpoint queue exceeds the predetermined length, remove an element from the head of the simulated checkpoint queue and increment the simulated write counter.

12. (Deleted).

13. The computer-readable storage medium of Claim 12, wherein the predetermined length being a dirty buffer limit.

14. The computer-readable storage medium of Claim 11 further storing computer instructions that, when executed by a computer, cause the computer to, write out of the any buffer from volatile memory and storing in nonvolatile memory using an incremental checkpoint operation in the normal operation of the database;

in response to detecting a write out of any buffer by the incremental checkpoint operation, check if that buffer is represented in the simulated checkpoint queue, and if that buffer is represented in the simulated checkpoint queue, remove the element representing that buffer from the simulated checkpoint queue and increment the simulated write counter.

15. (Deleted).

16. The computer-readable storage medium of Claim 11, wherein each element in the simulated checkpoint queue comprises:

a first identifier that identifies the respective buffer in the normal checkpoint queue;
and
a second identifier that identifies a journal entry in a redo log, the journal entry corresponding to the respective buffer.

17. The computer-readable storage medium of Claim 16, wherein the elements in the simulated checkpoint queue are ordered according to each element's journal entry position in the redo log.

18. The computer-readable storage medium of Claim 11 further storing computer instructions that, when executed by a computer, cause the computer to determine a dirty buffer limit for the simulated checkpoint queue, the dirty buffer limit specifying the length of the simulated checkpoint queue, the dirty buffer limit being determined from the simulated MTTR setting and historical operating data.

19. The computer-readable storage medium of Claim 18, wherein the historical operating data comprises an average time to read one journal entry in a redo log.

20. The computer-readable storage medium of Claim 18, wherein the historical operating data comprises an average time to read one buffer from nonvolatile memory to volatile memory.

21. A system for simulating effect of a recovery time required for the database system to recover from a database failure on runtime performance of the database system comprising:

- a memory for storing program instructions and data;

- one or more processors coupled to the memory;

- a simulated MTTR setting maintained in the memory;

- a normal checkpoint queue used in the database system for normal operation, wherein the normal checkpoint queue comprises a plurality of buffers;

- at least one simulated checkpoint queue;

- wherein the simulated checkpoint queue is an ordered list of one or more elements, each element in the simulated checkpoint queue representing a respective buffer that is or was in the normal checkpoint queue, the ordered list having a head and a tail;

- wherein the simulated checkpoint queue is associated with a setting for simulated mean time to recover (MTTR) whose effect on runtime performance of the database system is being simulated in the computer; and

- a simulated write counter maintained in the memory, the simulated write counter being associated with the simulated MTTR setting;

- wherein the simulated write counter provides a count of the number of times an element is removed from the simulated checkpoint queue, in response to a simulated write out of a buffer from volatile memory and storing in nonvolatile memory;

- wherein in response to detecting a change to a buffer in the normal checkpoint queue due to actual database transactions occurring within the database system under normal

operating conditions, the instructions in memory check if the buffer is represented in the simulated checkpoint queue, and if the buffer is not represented in the simulated checkpoint queue, the instructions link an element that represents the buffer to the tail of the simulated checkpoint queue;

wherein the instructions in memory determine if linking the element to the simulated checkpoint queue causes the simulated checkpoint queue to exceed a predetermined length; and

in response to determining that the simulated checkpoint queue exceeds the predetermined length, the instructions in memory remove an element from the head of the simulated checkpoint queue and increment the simulated write counter.

22. The system of Claim 21, wherein instructions in memory write out of the any buffer from volatile memory and store in nonvolatile memory using an incremental checkpoint operation in the normal operation of the database;

in response to detecting a write out of any buffer by the incremental checkpoint operation, instructions in memory check if that buffer is represented in the simulated checkpoint queue, and if that buffer is represented in the simulated checkpoint queue, remove the element representing that buffer from the simulated checkpoint queue and increment the simulated write counter.

23. (Deleted).

24. (Deleted).

25. (Deleted).

26. (Deleted).

27. The system of Claim 24, wherein the predetermined length being a dirty buffer limit, the dirty buffer limit being determined from the simulated MTTR setting and historical operating data.

28. The system of Claim 27, wherein the historical operating data comprises an average time to read one journal entry in a redo log.

29. The system of Claim 27, wherein the historical operating data comprises an average time to read one buffer from nonvolatile memory to volatile memory.

40. A computer-implemented method, to simulate effect on runtime performance of a database system, of a plurality of settings to control mean time to recover (MTTR) from a database failure, the method comprising:

using a current setting to limit MTTR of said database system for normal operation;
and

simulating performance of said database system for a plurality of additional settings of MTTR, using data from said normal operation.

41. The method of Claim 40, wherein said data comprises actual database operating data from said database system.

42. The method of Claim 40 wherein:
said simulating is performed concurrently with said performing of normal operation.

43. The method of Claim 42 wherein:
results of said simulating are available in real-time.

44. The method of Claim 42 wherein:
results of said simulating are available in quasi real-time.

45. The method of Claim 40 wherein:
said database system comprises a buffer in cache memory; and
said normal operation comprises a buffer change operation.

46. The method of Claim 40 further comprising:
said database system during said normal operation, maintaining a count of physical writes from volatile memory to nonvolatile memory that actually occur;
wherein said maintaining is simulated during said simulating performance, to maintain a plurality of simulated counters of said physical writes that would have occurred for corresponding additional settings of MTTR being simulated.

47. The method of Claim 40 further comprising:
said database system during said normal operation, using a normal checkpointing queue; and
said simulating using a plurality of simulated checkpointing queues for corresponding additional settings of MTTR being simulated.

48. A computer-readable storage medium having stored thereon computer instructions that, when executed by a computer, cause the computer to, simulate effect on runtime performance of a database system, of a plurality of settings to control mean time to recover (MTTR) from a database failure, the computer instructions comprising instructions to:
use a current setting to limit MTTR of said database system for normal operation; and
simulate performance of said database system for a plurality of additional settings of MTTR, using data from said normal operation.

49. The computer-readable storage medium of Claim 48, wherein said data comprises actual database operating data from said database system.

50. The computer-readable storage medium of Claim 48, wherein:
said instructions to simulate are executed concurrently with said instructions to perform normal operation.

51. The computer-readable storage medium of Claim 50 wherein:

results of simulating are available in real-time.

52. The method of Claim 50 wherein:

results of simulating are available in quasi real-time.

53. The computer-readable storage medium of Claim 48 wherein:

said database system comprises a buffer in cache memory; and

said normal operation comprises a buffer change operation.

54. The computer-readable storage medium of Claim 48 further comprising

instructions for:

said database system during said normal operation, to maintain a count of physical writes from volatile memory to nonvolatile memory that actually occur;

wherein said maintaining of the count is simulated by said instructions to simulate performance, to maintain a plurality of simulated counters of said physical writes that would have occurred for corresponding additional settings of MTTR being simulated.

55. The computer-readable storage medium of Claim 48 further comprising

instructions for:

said database system during said normal operation, to use a normal checkpointing queue; and

wherein said instructions to simulate performance comprise instructions to use a plurality of simulated checkpointing queues for corresponding additional settings of MTTR being simulated.

56. A system, to simulate effect on runtime performance of a database system, of a plurality of settings to control mean time to recover (MTTR) from a database failure, the system comprising:

means for using a current setting to limit MTTR of said database system for normal operation; and

means for simulating performance of said database system for a plurality of additional settings of MTTR, using data from said normal operation.

57. The system of Claim 56, wherein said data comprises actual database operating data from said database system.

58. The system of Claim 56 wherein:

said means for simulating operates concurrently with said normal operation of said database system.

59. The system of Claim 58 wherein:

results of said simulating are available in real-time.

60. The method of Claim 58 wherein:

results of said simulating are available in quasi real-time.

61. The system of Claim 56 wherein:

said database system comprises a buffer in cache memory; and

said normal operation comprises a buffer change operation.

62. The system of Claim 56 further comprising:

means for maintaining a count of physical writes from volatile memory to nonvolatile memory that actually occur during said normal operation;

wherein said maintaining of the count is simulated in said means for simulating performance, to maintain a plurality of simulated counters of said physical writes that would have occurred for corresponding additional settings of MTTR being simulated.

63. The system of Claim 56 further comprising:

means for using a normal checkpointing queue during said normal operation;

wherein said means for simulating performance comprises means for using a plurality of simulated checkpointing queues for corresponding additional settings of MTTR being simulated.